SCREEN CONTENT IMAGE SEGMENTATION USING LEAST ABSOLUTE DEVIATION FITTING

Shervin Minaee and Yao Wang

Department of Electrical and Computer Engineering, Polytechnic School of Engineering,
New York University, NY, USA.

ABSTRACT

We propose an algorithm for separating the foreground (mainly text and line graphics) from the smoothly varying background in screen content images. The proposed method is designed based on the assumption that the background part of the image is smoothly varying and can be represented by a linear combination of a few smoothly varying basis functions, while the foreground text and graphics create sharp discontinuity and cannot be modeled by this smooth representation. The algorithm separates the background and foreground using a least absolute deviation method to fit the smooth model to the image pixels. This algorithm has been tested on several images from HEVC standard test sequences for screen content coding, and is shown to have superior performance over other popular methods, such as k-means clustering based segmentation in DjVu and shape primitive extraction and coding (SPEC) algorithm. Such background/foreground segmentation are important pre-processing steps for text extraction and separate coding of background and foreground for compression of screen content images.

1. INTRODUCTION

Screen content images refer to images appearing on the display screens of electronic devices. These images share similar characteristics as mixed content documents (such as a magazine page). They often contain two layers, pictorial background and the foreground consisting of text and line graphics. The usual image compression algorithm such as JPEG2000 [1] and HEVC intra frame coding [2] may not result in a good compression rate for this kind of images. In these cases, segmenting the image into two layers and coding them separately may be more efficient. The idea of segmenting an image for better compression was proposed for check image compression [3], in DjVu algorithm for scanned document compression [4] and the mixed raster content representation [5].

Screen content images are hard to segment, because the foreground may be overlaid over a smoothly varying background that has a color range that overlaps with the color of the foreground. Also because of the use of sub-pixel rendering, the same text/line often has different colors. Even in the absence of sub-pixel rendering, pixels belonging to the same text/line often has somewhat different colors.

Most of previous works regarding foreground segmentation are based on color clustering or color counting and have difficulty for cases where the background color has a large dynamic range or is similar to the foreground color in some regions. The hierarchical k-means clustering algorithm initially proposed in DjVu [4] is a representative algorithm based on color clustering. This algorithm applies the k-means clustering algorithm with k=2 on blocks in multi-resolution. It first applies the k-means clustering algorithm on a large block to obtain foreground and background colors and then uses them as the initial foreground and background colors for the smaller blocks in the next stages. This algorithm has difficulty for the regions where background and foreground color intensities overlap.

In the shape primitive extraction and coding (SPEC) method, which was developed for segmentation of screen content [6], a two-step segmentation algorithm is proposed. In the first step they classify each block of size 16×16 into either pictorial block or text/graphics based on the number of colors. If the number of colors is more than a threshold, it will be classified into pictorial block, otherwise to text/graphics. In the second step, they refine the segmentation result of pictorial blocks, by extracting shape primitives (horizontal line, vertical line or a rectangle with the same color) and then comparing the size and color of the shape primitives with some threshold. Because blocks containing smoothly varying background over a narrow range can also have a small color number, it is hard to find a fixed color number threshold that can robustly separate pictorial blocks and text/graphics blocks. Furthermore, text and graphics in screen content images typically have some variation in their colors, even in the absence of sub-pixel rendering. These challenges limit the effectiveness of SPEC.

In this paper, we propose a foreground/background separation algorithm which uses the smoothness property of the background and the fact that the foreground pixels typically deviate greatly from the smooth function fit to the background. Using this intuition we propose to use least absolute deviation approach [7] to fit each image block using a smooth model. Those pixels which can be represented with small

distortion using this smooth model will be considered as background and the rest as foreground. This technique can be used for screen content video coding [8], text extraction [9], medical image segmentation and classification [10], [11] and principal line extraction from palmprint images [12], [13].

2. LEAST ABSOLUTE DEVIATION

In this paper, we look at the foreground segmentation problem from signal decomposition point of view. We assume that the background part of the image can be well represented with a simple smooth model, whereas the foreground pixels cannot be represented accurately with this smooth model. By well representation we mean that the distortion between the approximated smooth model and the actual pixel values is less than a desired threshold. To be more specific, we divide each image into non-overlapping blocks of size N×N, and then represent each image block denoted by $F(x, y)$, with a smooth model $S(x, y; \alpha_1, ..., \alpha_K)$, where x and y denote the horizontal and vertical axes and $\alpha_1, ..., \alpha_K$ denote the parameters of this smooth model. For color images, $F(x, y)$ represents the luminance component. In order to find the optimal model parameters, α_k's, we should define some cost function which measures the goodness of fit between the intensity of background pixels in the original image and the one predicted by smooth model, and then minimize the cost function as:

$$\{\alpha_1^*, ..., \alpha_K^*\} = \arg \min_{\alpha_1, ..., \alpha_K} \|F(x, y) - S(x, y; \alpha_1, ..., \alpha_K)\|_p$$

Now two questions should be answered:

1. What is an optimal smooth model for background layer representation.

2. What error measure (i.e. p value) to use such that the model parameters are mainly found using the background pixels.

For the first question, we propose to use a linear combination of K smooth basis functions $\sum_{k=1}^{K} \alpha_k P_k(x, y)$, where $P_k(x, y)$ denotes a 2D smooth basis function. Here we use a set of low frequency two-dimensional DCT basis functions, since they have been shown to be very efficient for image representation [14]. The 2-D DCT function is defined as:

$$P_{u,v}(x, y) = \beta_u \beta_v cos((2x+1)\pi u/2N) cos((2y+1)\pi u/2N)$$

where u and v denote the frequency of the basis. We order all the possible basis functions in the conventional zig-zag order in the (u,v) plane, and choose the first K basis functions. We have found that K=10 leads to very good background representation for a variety of screen content images (with PSNR over 45dB), and additional bases do not lead to significant increase in the reconstruction quality.

Using this linear model, we need to solve the following optimization problem to derive model parameters:

$$\{\alpha_1^*, ..., \alpha_K^*\} = \arg \min_{\alpha_1, ..., \alpha_K} \|F(x, y) - \sum_{k=1}^{K} \alpha_k P_k(x, y)\|_p$$

We can also look at the 1D version of this problem by converting the 2D blocks of size $N \times N$ into a vector of length N^2, denoted by f, by concatenating the columns and denoting $\sum_{k=1}^{K} \alpha_k P_k(x, y)$ as $P\alpha$ where P is a matrix of size $N^2 \times K$ in which the k-th column corresponds to the vectorized version of $P_k(x, y)$ and $\alpha = [\alpha_1, ..., \alpha_K]^T$. Then the problem can be formulated as: $\alpha^* = \text{argmin}_\alpha \|f - P\alpha\|_p$

For the second question, we can use different distances between actual pixel values and approximated ones with the smooth model. As an example, by minimizing the l2 norm we will have the least-sqaure fitting problem. The least square fitting suffers from the fact that the model parameters, α, can be adversely affected by foreground pixels. In least-square fitting, by squaring the residuals, the larger residues will get larger weights in determining the model parameters. Because of that we propose to use least absolute deviation, which is more robust to outliers compared to least-square fitting and the model is less affected by outliers. Therefore we need to solve the following optimization problem:

$$\alpha^* = \text{argmin}_\alpha \|f - P\alpha\|_1 \qquad (1)$$

Least absolute deviation problem does not have a closed form solution but it can be solved with iterative algorithms. Different algorithms can be used to solve this problem, such as alternating direction method of multipliers (ADMM) [15], iterative reweighted least square fitting [16] and linear programming. Here we use the ADMM algorithm.

One alternative way to solve the second question is to use a robust regression approach to fit the smooth model into image blocks such that the model parameters are determined only using background pixels. One such a work is presented in [17] where the author proposed to use RANSAC algorithm to fit the smooth model into the background pixels.

2.1. ADMM formulation to solve least absolute deviation

To solve (1) with ADMM, we introduce the auxiliary variable $z = P\alpha - f$ and convert the original problem into the following form:

$$\begin{aligned} \underset{z, \alpha}{\text{minimize}} \quad & \|z\|_1 \\ \text{subject to} \quad & P\alpha - z = f. \end{aligned}$$

Then we can use the following updates for each iteration in ADMM [15]:

$$\alpha^{k+1} = (P^T P)^{-1} P^T (f + z^k - u^k)$$
$$z^{k+1} = S_{1/\rho}(P\alpha^{k+1} - f + u^k)$$
$$u^{k+1} = u^k + P\alpha^{k+1} - z^{k+1} - f$$

where u denotes the dual variable, ρ is the augmented Lagrangian parameter and $S_{1/\rho}$ denotes soft-thresholding operator applied elementwise and is defined as:

$$S_{1/\rho}(x) = \text{sign}(x)\max(|x| - 1/\rho, 0)$$

2.2. Segmentation Algorithm

We propose a segmentation algorithm which first checks if a block can be segmented using some simpler methods. These simple cases take care of two groups of blocks: completely flat block and smoothly varying background without foreground. Completely flat blocks are those in which all pixels have the same value and are common in screen content images. Therefore they can be declared as background or foreground based on their neighboring blocks' background color. For these blocks, if we could find at least one neighbor block with a background color close enough to the current block's color (difference less than ϵ_2), it would be segmented as background. Smoothly varying background without foreground is a block in which the intensity of all pixels can be modeled well by the smooth function. Therefore we try to fit K DCT basis to all pixels using least square fitting and if the intensity of all pixels can be predicted with distortion less than ϵ_3, that block would be segmented entirely as background. We will apply the least absolute fitting only if a block does not satisfy these two conditions. Furthermore, at the end of the least absolute fitting, we check the percentage of identified background pixels. If the percent is less than a threshold, we divide the block into 4 smaller blocks and repeat the process. The overall segmentation algorithm is summarized below:

1. If all pixels in the block have the same color intensity (i.e. it is completely flat block), declare the entire block as background or foreground as explained above. If not, go to the next step;

2. Perform least square fitting using the luminance values of all pixels. If all pixels can be predicted with an absolute error less than ϵ_3, declare the entire block as background. If not, go to the next step;

3. Use least absolute deviation to fit a model to the luminance values of image block and find the absolute fitting error of all pixels using that model. Each pixel with a distortion less than a threshold ϵ_1 will be considered as background, otherwise as foreground. If the percentage of background pixels is more than ϵ_4 then stop, otherwise go to the next step;

4. Decompose current block of size $N \times N$ into 4 smaller blocks of size $\frac{N}{2} \times \frac{N}{2}$ and run the segmentation algorithm for all of them. Repeat until $N = 8$.

The above algorithm makes an initial decision based on the luminance component of a block only. At the end of this process, we further use least squares fitting to find a smooth model for each of the two chrominance components (Cb and Cr) using the chrominance values at identified background pixels. If the fitting error for any color component is larger than ϵ_1 for any pixel, that pixel is reclassified as foreground.

3. RESULTS

To enable rigorous evaluation of different algorithms, we have generated an annotated dataset consisting of 332 image blocks of size 64×64, extracted from sample frames from 5 HEVC test sequences for screen content coding. The ground truth foregrounds for these images are extracted manually.

Before showing the results let us discuss about parameters of our algorithm. In our implementation, the block size is chosen to be N=64 which is the same as the largest CU size in HEVC standard. The number of DCT basis functions, K, is chosen to be 10 based on the training images. The other parameters are chosen as $\epsilon_1 = 10$, $\epsilon_2 = 10$, $\epsilon_3 = 3$ and $\epsilon_4 = 0.5$, which have been found to perform well on a training set. For ADMM algorithm, we have used the implementation by Stephen Boyd [18]. The number of iteration is chosen to be 200 and the parameter ρ is chosen as the default value, 1.

We compare the proposed algorithm with two algorithms; hierarchical k-means clustering in DjVu and shape primitive extraction and coding (SPEC) method. For SPEC, we have adapted the color number threshold and the shape primitive size threshold from the default value given in [6] when necessary to give more satisfactory result. Furthermore, for blocks classified as text/graphics based on the color number, we segment the most frequent color and any similar color to it (i.e. colors which their distance from most frequent color is less than 10) in the current block as background and the rest as foreground.

To provide a numerical comparison between the proposed scheme and previous approaches, we have calculated the average precision and recall achieved by different segmentation algorithms over this dataset. The average precision and recall by different algorithms are given in Table 1. As it can be seen, the proposed scheme achieves a much higher precision and recall than other algorithms.

Table 1: Accuracy comparison of different algorithms

Performance Criteria	SPEC	Clustering in DjVu	The proposed algorithm
Precision	0.5038	0.6491	0.9147
Recall	0.6458	0.6909	0.8773

The results for 5 test images (each consisting of multiple 64x64 blocks) are shown in Fig. 1. It can be seen that in all cases the proposed algorithm gives superior performance over DjVu and SPEC. Note that our dataset mainly consists of challenging images where the background and foreground
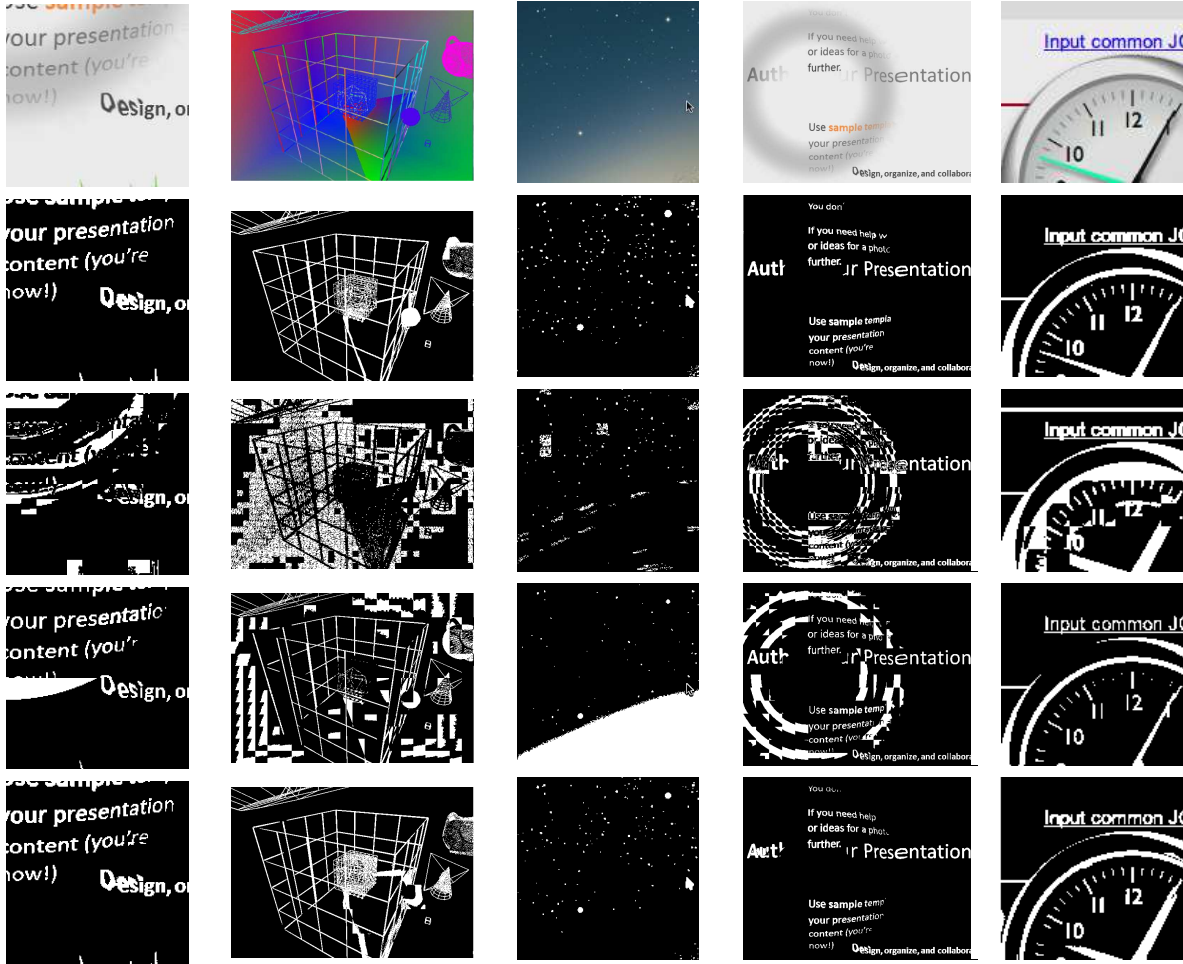
Fig. 1: Segmentation result for test images. The images in the first and second rows denote the original images and ground truth foregrounds. The images in the third, forth and the fifth rows denote the foreground map by shape primitive extraction and coding, hierarchical clustering in DjVu and the proposed algorithm respectively.

have overlapping color ranges. For simpler cases where the background has a narrow color range that is quite different from the foreground, both DjVu and the proposed method will work well. On the other hand, SPEC does not work well when the background is fairly homogeneous within a block and the foreground text/lines have varying colors.

4. CONCLUSION

This paper proposed an algorithm for segmentation of screen content images into a foreground layer consisting of mainly text and lines and a background layer consisting of smoothly varying regions. We developed a least absolute deviation approach to fit a smooth model into image blocks. A pixel is considered background if it can be represented accurately by the smooth model; otherwise it will be considered as foreground. Instead of applying this algorithm to every block, which is computationally demanding, we first check whether a block can be segmented using simpler methods. This helps to reduce the computation complexity. This algorithm has

been tested on several test images and compared with two well-known algorithms for foreground segmentation, SPEC and hierarchical clustering in DjVu, and it shows significantly better performance for blocks where the background and foreground pixels have overlapping intensities. Note that the proposed algorithm is not limited to screen content image segmentation. It has other applications such as text extraction in images and principal line extraction in palmprint images.

5. REFERENCES

[1] A. Skodras, C. Christopoulos and T. Ebrahimi, "The JPEG 2000 still image compression standard", IEEE Signal Processing Magazine, (2001): 36-58.

[2] G.J. Sullivan, J. Ohm, W.J. Han and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard", IEEE Transactions on Circuits and Systems for Video Technology, 22(12), (2012): 1649-1668.

[3] J. Huang, EK. Wong and Y. Wang, "Check image compression using a layered coding method", Journal of Electronic Imaging 7.3 (1998): 426-442.

[4] P. Haffner, P.G. Howard, P. Simard, Y. Bengio and Y. Lecun, "High quality document image compression with DjVu", Journal of Electronic Imaging, 7(3), 1998, 410-425.

[5] R.L. DeQueiroz, R.R. Buckley and M. Xu, "Mixed raster content (MRC) model for compound image compression", Electronic Imaging'99. International Society for Optics and Photonics, 1998.

[6] T. Lin and P. Hao, "Compound image compression for real-time computer screen image transmission", IEEE Transactions on Image Processing, 14(8), 2005, 993-1005.

[7] Y. Li and G.R. Arce, "A maximum likelihood approach to least absolute deviation regression", EURASIP Journal on Applied Signal Processing, 2004, 1762-1769.

[8] M. Zhan, X. Feng and M. Xu, "Advanced screen content coding using color table and index map", IEEE International Conference on Multimedia and Expo Workshops, 2014.

[9] J. Zhang and R. Kasturi, "Extraction of Text Objects in Video Documents: Recent Progress", Document Analysis Systems. 2008.

[10] S. Minaee, M. Fotouhi and B.H. Khalaj, "A geometric approach for fully automatic chromosome segmentation", arXiv preprint arXiv:1112.4164, 2011.

[11] U. Srinivas, H. Mousavi, C. Jeon, V. Monga, A. Hattel and B. Jayarao, "SHIRC: A simultaneous sparsity model for histopathological image representation and classification", IEEE International Symposium on Biomedical Imaging, 2013.

[12] SA. Mistani, S. Minaee and E. Fatemizadeh, "Multi-spectral palmprint recognition using a hybrid feature", arXiv preprint arXiv:1112.5997 (2011).

[13] S. Minaee and AA. Abdolrashidi, "Multispectral palmprint recognition using textural features", Signal Processing in Medicine and Biology Symposium (SPMB), 2014 IEEE.

[14] A.B. Watson, "Image compression using the discrete cosine transform", Mathematica journal 4.1 (1994): 81.

[15] S. Boyd, N. Parikh, E. Chu, B. Peleato and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers", Foundations and Trends in Machine Learning, 3(1), 2011, 1-122.

[16] I. Daubechies, R. DeVore, M. Fornasier and C.S. Gunturk, "Iteratively reweighted least squares minimization for sparse recovery", Communications on Pure and Applied Mathematics, 63(1), 2010, 1-38.

[17] S. Minaee, H. Yu and Y Wang, "A Robust Regression Approach for Background/Foreground Segmentation", arXiv preprint arXiv:1412.5126 (2014).

[18] https://web.stanford.edu/ boyd/papers/admm/